

REMARKS

Reconsideration of this application, in view of the foregoing amendments and the following remarks, is requested.

Claims 29-52 were presented for consideration in this application. By the foregoing amendment, claims 29 and 37 are amended to correct grammatical errors. Claims 29-52 are now pending.

Rejections under 35 U.S.C. § 103

Claims 29-52 are rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,567,887 (“Harmer”) in view of U.S. Patent No. 6,604,170 (“Suzuki”). This rejection is traversed.

Claim 29 recites, in part,

opening the file using a file open operation comprised in a file metadata processing module loaded in a shared execution portion of the secure RAM, wherein the file open operation traverses a file access table (FAT) of the file system to determine a sequence of clusters allocated to the file and stores a cluster identifier for each cluster in the sequence in a buffer comprised in a shared data portion of the secure RAM, wherein the cluster identifiers are stored in the buffer such that each cluster identifier is locatable by an index computed using a cluster size and a start offset of data in the file

The Examiner relies on Suzuki, col. 9, ll. 15-25, to teach “the file open operation traverses a file access table (FAT) of the file system to determine a sequence of clusters allocated to the file.” While the portion of Suzuki relied on by the Examiner does disclose traversal of a file access table, the traversal is performed as part of reading a file, not as part of opening a file as required by claim 29. See Suzuki, col. 9, ll. 13-53.

Further, even assuming arguendo that Suzuki teaches what the Examiner asserts, the Examiner has still not shown that the combination of Harmer and Suzuki teaches the above recited limitations of claim 29. Claim 29 clearly requires that a file open operation stores in a buffer a cluster identifier for each cluster in the sequence of clusters identified when the FAT is traversed. The Examiner relies on Harmer, col., 6, ll. 18-22 and ll. 25-35 to allegedly teach these limitations, stating that these portions of Harmer disclose “clusters

of files stored in FAT table which is cached in RAM.” Applicant notes that there is no mention of any type of file open operation in the portions of Harmer relied on by the Examiner, much less a file open operation that stores cluster identifiers in a buffer. In fact, when these portions are taken in context, it is clear that Harmer is disclosing information related to reads and writes of files, not opening of files. See Harmer, col. 6, ll. 12-23.

In addition, as best Applicant can discern given the paucity of explanation provided by the Examiner, the Examiner is erroneously interpreting the disclosure of Harmer that a FAT table is “cached” in RAM as somehow teaching that only part of a FAT table is stored in RAM and as the FAT table is accessed to locate clusters of files, other portions of the FAT table may be brought into the RAM, thus causing storage of cluster identifiers in the FAT table in the RAM. Such an interpretation is not supported by Harmer. Harmer specifically discloses that all file system information such as directory information and cluster information is stored in RAM and not on the disk. For example, Harmer specifically states that “[w]ith the present invention, it is no longer necessary to go to the drive to determine the location of the data to be written or read from the disk.” Harmer, col. 6, ll. 12-14. Further, note that in Fig. 2, the directory information 206, cluster information 204, and sector information 202 are depicted as being stored in the system RAM 50 and not on the mass storage peripheral device 56.

Moreover, while Harmer discloses that a “caching mechanism manages a portion of host memory for caching file system data,” there appears to be no disclosure in Harmer that this caching mechanism stores only a part of the file system information in RAM and updates the file system information from the disk if the information needed is not in the RAM. Harmer, col. 7, ll. 27-28. Any attempt to infer such a disclosure would be in direct opposition to the above quotation from Harmer stating that that with the invention disclosed in Harmer, there is no need to go to the disk to access file system information. Also note that Fig. 10 shows that the file system information (1002, 1006, 1008) is “fixed in memory.” Thus, Harmer does not disclose a file open operation that stores in a buffer a cluster identifier for each cluster in the sequence of clusters identified when the FAT is traversed as required by claim 29.

Claim 29 further requires that “the cluster identifiers are stored in the buffer such that each cluster identifier is locatable by an index computed using a cluster size and a start offset of data in the file.” The Examiner asserts that Harmer, col. 6, ll. 14-18, discloses these limitations. This portion of Harmer merely states, in its entirety, that “[i]t is only necessary for the loadable device driver 66 to access or query the caching mechanism 210 in order to determine this information.” A mere mention of accessing or querying a caching mechanism that manages file system information cannot possibly be read to disclose the specific storage requirements for cluster identifiers cited in claim 291. Thus, the Examiner has not shown that Harmer discloses the cited limitations of claim 29.

Claim 29 also recites “accessing the file using a file access operation comprised in a file data processing module loaded in the shared execution portion, wherein the data processing module overlays at least a portion of the metadata processing module.” The Examiner asserts that Harmer, col. 6, ll. 6-11, discloses these limitations. The Examiner states that this portion of Harmer discloses a “caching mechanism for FAT table” and “a cache inherently does not have the capacity to hold the entirety of a structure such as a hard drive.” As best Applicant can tell given the paucity of explanation provided by the Examiner, the Examiner is erroneously equating the file metadata processing module of claim 1 to a FAT table as disclosed in Harmer and the file data processing module of claim 1 to the caching mechanism disclosed in Harmer. The Examiner is also erroneously attributing “inherent” functionality to the caching mechanism disclosed by Harmer that allows the caching mechanism to store only a portion of the FAT table in RAM and to bring other portions of the FAT table, presumably from a disk, into RAM and possibly overwrite portions of the FAT table already in RAM. Such an interpretation is not supported by Harmer.

The Examiner’s attribution of inherent functionality to the caching mechanism is in direct opposition to the teachings of Harmer. Harmer specifically states that “[w]ith the present invention, it is no longer necessary to go to the drive to determine the location of the data to be written or read from the disk.” Harmer, col. 6, ll. 12-14. Further, note that in Fig. 2, the directory information 206, cluster information 204, and sector information 202

are depicted as being stored in the system RAM 50 and not on the mass storage peripheral device 56. In addition, Harmer specifically states that “the caching mechanism 210 of loadable device driver 66 manages the storage, lookup and updating of the file system information directly out of system RAM 50. Harmer, col. 4, ll. 35-38. Also note that Fig. 10 shows that the file system information (1002, 1006, 1008) is “fixed in memory.” Thus, the Examiner has not shown that Harmer discloses the cited limitations of claim 29.

Independent claims 37 and 45 include limitations similar to those discussed above for claim 29 and thus are patentable over the cited prior art for at least the same reasons. The dependent claims are also patentable over the cited prior art for at least the same reasons. Accordingly, withdrawal of the rejection is requested.

Conclusion

Applicant believes this application and the claims herein to be in a condition for allowance and respectfully requests that the Examiner allow this application to pass to the issue branch.

Applicant believes that no additional fee is due at this time; however, please charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 relating to this matter to Deposit Account Number 20-0668, for Texas Instruments Incorporated.

Should the Examiner have further inquiry concerning these matters, please contact the below named attorney for Applicant.

Respectfully submitted,

/Ellen Baker Laws/

Ellen Baker Laws
Attorney for Applicant
Reg. No. 50272
713-937-8823

Texas Instruments Incorporated
P.O. Box 655474, MS 3999
Dallas, TX 75265
(972) 917-5287